

SciPass: a 100Gbps capable secure Science DMZ using OpenFlow and Bro

Edward Balas
GlobalNOC
Indiana University
Bloomington, IN 47408
ebalas@iu.edu

AJ Ragusa
GlobalNOC
Indiana University
Bloomington, IN 47408
aragusa@iu.edu

ABSTRACT

In this paper, we describe a 100Gbps capable OpenFlow based Science DMZ approach which combines adaptive IDS load balancing, dynamic traffic filtering and a novel IDS based technique to detect “good” traffic flows and forward around performance challenged institutional firewalls. Evaluation of this approach was conducted using GridFTP and Iperf3. Results indicate this is a viable approach to enhance science data transfer performance and reduce security hardware costs.

Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design; C.2.3 [Computer-Communication Networks]: Network Operations—*network management, network monitoring*; C.2.5 [Computer-Communication Networks]: Local and Wide-Area Networks—*internet*

General Terms

Design, Management, Measurement, Performance, Security

Keywords

Keywords are your own designated keywords.

1. INTRODUCTION

Research institutions engaged in data-intensive science often find that inadequate campus cyberinfrastructure prevent their researchers from fully benefiting from existing high performance research and education network capacity[1]. To address these cyberinfrastructure shortcomings, the Dart et al proposed the Science DMZ network design pattern.

Central to this pattern is the recognition that some components in modern campus network such as institutional firewalls are designed to support a large number of small traffic flows rather than the small number of large flows often seen in data-intensive science. The challenge is to mitigate the negative performance impact of these components without degrading the security of the infrastructure.

Too often network security and network performance are assumed to be diametrically opposed. In this paper we outline an approach to augment the Science DMZ concept with a 100Gbps capable Intrusion Detection System cluster which would not only offer potentially improved security but would also provide a basis to identify “good” science data transfers and provide an enhanced bandwidth experience. The results of our efforts to date in developing a system called SciPass[2], which extends the Science DMZ concept to contain such an IDS cluster and ability to reactively bypass institutional firewalls are described.

The SciPass system has its origins in a prior project at Indiana University called FlowScale[3]. The goal of FlowScale project was to create a cost effective Intrusion Detection System(IDS) load balancer based on an SDN substrate. It employed an OpenFlow[4] switch and custom controller to divide campus traffic across a cluster of IDS sensors. The primary advantage of this approach was the ability to use a standard switch versus a dedicated appliance to perform the balancing task. Based on two years of operating this production at Indiana University, the SciPass system was conceived as an evolution of FlowScale.

The first goal of this new system was to create a feedback loop between the controller and the set of IDS sensors, with each sensor reporting its current load periodically. This allows the balancing routine factor these non-network metrics when which traffic to send to a particular sensor.

The second goal was to provide a means for the IDS sensors to identify flows that were uninteresting from a security standpoint and signal that those flows should not be sent to the sensors, in effect creating a white list capability. The impact of this feature is a significant reduction in workload for the sensors.

Normally, IDS are used to detect hostile or bad flows, however they are equally adept at detecting institutionally valuable flows such as science data transfers. The third goal was to extend the capability of the white lists by forwarding good flows around low performing components such as firewalls. SciPass looks for “good” flows such as large science data transfers and, using OpenFlow, bypasses the firewall.

2. Approach

The SciPass system contains 5 components: an OpenFlow Switch, the SciPass controller, a cluster of IDS sensors, a PerfSONAR host, a firewall and a Data Transfer Node(DTN). SciPass today uses the Bro Intrusion Detection System[5] for each sensor in the sensor cluster.

SciPass defines IDS policy to identify “good” flows. These policies contain a combination of time of the day and day of the week, source and designation IP address along with protocol and application layer data to determine if a flow should bypass an institutional firewall.

Imagine for instance that a scientist uploads genomic data to the same facility across the country every Friday from a local DTN. SciPass could be configured to only bypass the firewall when transfers out of the directory “/data/genomics/project-x/” to the specific remote facility on Friday between 2 and 8 am. In this way the policy gives scientists, network administrators and

security administrators the ability to jointly define and enforce desired network behavior.

By default, traffic is forwarded through the OpenFlow switch via the institutional firewall. As this happens, copies of packets are sent to the array of IDS sensors. SciPass uses a balancing mechanism that ensures that all packets for a given flow go to the same sensor for stream reassembly and that flows are distributed as evenly as possible across the array of sensors. Using this approach lets one monitor individual 100Gbps network connections using an array of 1 or 10Gbps capable IDS sensors.

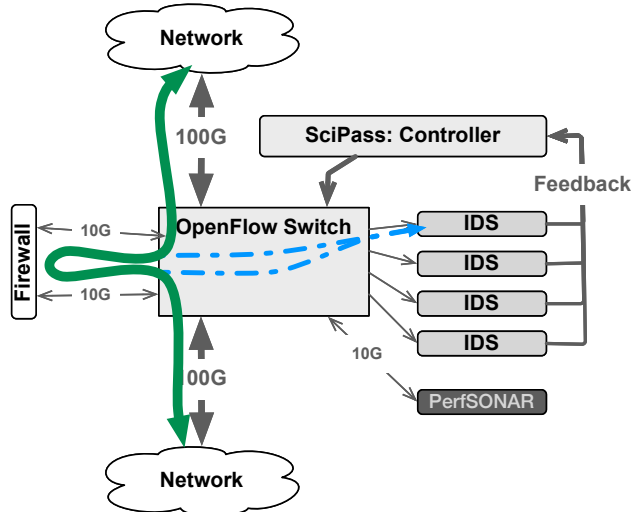


Figure 1 Default forwarding through Firewall with packets copied to 1 of the IDS sensors.

When the system determines it is appropriate to bypass an individual flow around the institutional firewall, a pair of higher priority OpenFlow rules is added to the switch so that packets associated with this flow are directly forwarded from the North port to the South port on the OpenFlow switch, bypassing the default path which includes the institutional firewall and the IDS array. These rules contain an idle timeout such that once the flow completes the rules will be purged from the switch.

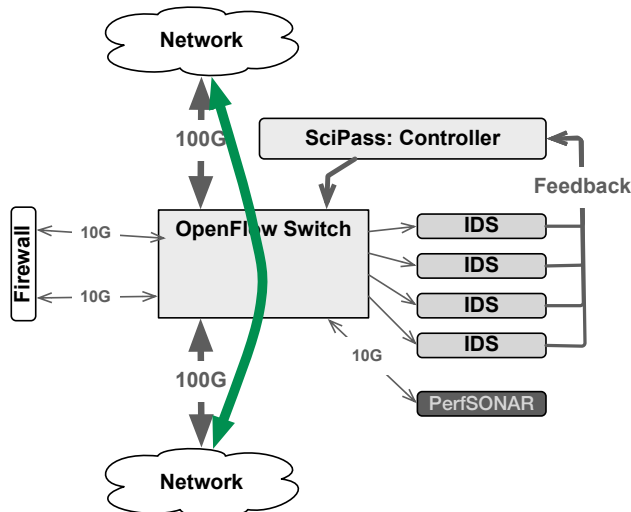


Figure 2 Science Flows programmed to bypass firewall

We expect this approach will have two benefits. First, science data users will see dramatically improved transfer performance as inadequate cyberinfrastructure is removed from the forwarding path. Second, campus security and network operators will be able to provide 100Gbps security with lower costs by not sending known good traffic through institutional firewalls and IDS clusters.

3. Methodology

For all tests conducted in this paper, all lab components were interconnected with 10GbE interfaces. A Brocade MLXe-4 running version 5.60d in layer23 mode was used as the OpenFlow switch. The SciPass controller was locally connected to the switch providing < 1ms RTT between the controller and switch. A Netscreen 5200 was used as an example of an institutional firewall. It contained 2 x 10GbE interfaces, was running the latest code revision and was configured with the best known tunings using a default open policy to forward all traffic.

GridFTP[6] and iperf3[5] were used to evaluate network performance. 2 DTNs were located in Indiana University's InCNTRE lab for low latency testing. In addition, we used a public test DTN hosted by ESnet at Argonne National Laboratory. The remote DTN at Argonne had a 7ms Round Trip Time from our lab DTNs. The path crossed 5 organizational boundaries and contained a combination of 10Gbps and 100Gbps links.

4. Firewall Impact on Transfer Performance

Much of the motivation for the DMZ architecture hinges on the notion that institutional firewalls are designed for a large number of small flows and do not support use cases involving large flows. Our first task was to test this notion under more controlled circumstances than are typically found in production.

4.1 Bypass Evaluation

To measure host performance, a single flow TCP data transfer using iperf3 was conducted for 10 seconds with 2 hosts directly connected to the OpenFlow Switch. The flow achieved an average bandwidth of 9.9Gbits/sec with no retransmissions. This performance was consistent with that of modern well performing hosts and represents a baseline for comparison, which represents the performance to expect when a flow is bypassing the firewall

4.2 Firewall Evaluation

To measure the firewall performance, the same test was conducted but with the firewall in the forwarding path. In this test the flow achieved an average bandwidth 1.30 Gbits/sec with 2838 retransmissions. Recorded packet captures contained duplicate ack and retransmissions consistent with packet loss. None of the interface counters on the switch, firewall or hosts indicated errors, however one observation implicated the firewall. Closer examination of the packet counters on the firewall revealed that it transmitted fewer packets to the destination host than it received. Because test traffic was the only traffic on the firewall, this clearly indicated that the source of loss was inside the firewall. This performance was also consistent with our expectation.

4.3 Firewall with Latency

To measure impact of loss over higher latency paths, we emulated the latency we observed in testing against a GridFTP server at Argonne National Laboratory. From the location of our lab, the Argonne server had a RTT of 7ms. The Linux utility TC was used on the receiving host to introduce the equivalent amount of delay into the lab. In this test, we saw 117Mbps/sec over the 10 seconds with 119 retransmissions. This was consistent with expectation.

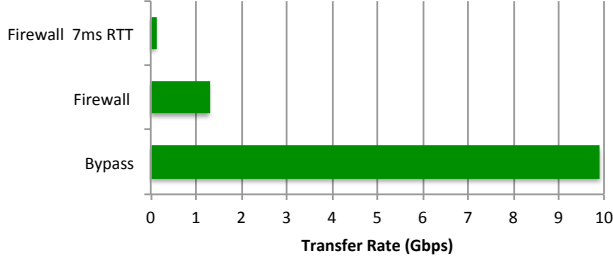


Figure 3 transfer rate across different forwarding paths

5. SciPass Performance Impact

When designing SciPass, a key concern to address was how quickly the system could detect and program the bypass forwarding rules. The amount of time involved in this task has a direct impact on how broadly applicable this technique is to network flows typically found on a campus network.

To evaluate the impact the system's reaction latency may have on data transfer performance, we compared three GridFTP transfers performed from Indiana University to a server at Argonne National Laboratory, in all 3 cases the latency was 7ms. Both servers in the test were using Hamilton TCP[8]. In the first test, we evaluated transfer performance through the firewall. In the second test, we manually preconfigured a bypass before the transfer started. For the last test we waited 8 seconds before switching to the firewall bypass path. Eight seconds was selected as it represented what we thought might be a worst possible case for reactively programming a bypass forwarding path.

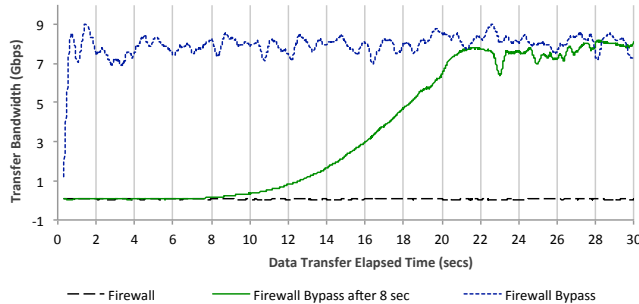


Figure 4 data transfer with manual bypass after 8 seconds

The firewall test performance was inline with our prior lab tests with 7ms of emulated latency. The proactive firewall bypass case indicates a rapid bandwidth growth to ~ 9 Gbps within 1 second consistent with normal slow start phase of a TCP session. In our

reactive use case, we are effectively providing additional bandwidth after the slow start phase has concluded and the session is in congestion avoidance. In this case, the result is the session taking more than 12 seconds to fully utilize the additional bandwidth.

Next, we observed the performance of SciPass itself to determine how long it would normally take to program a bypass. SciPass reactions contain three phases. First, the Bro IDS monitors GridFTP sessions extracting information about the data transfer sessions and, based on policy, signals the SciPass controller. Second, the SciPass controller, using OpenFlow, requests a set of switch forwarding table modifications to bypass the firewall. Third, the switch installs these new rules into the hardware based forwarding tables on each line card. In our testing, the total amount of time for the SciPass system to detect a GridFTP session and redirect traffic around the firewall was 64ms. This indicated that it might be possible to address the firewall bottleneck before the TCP session had significant loss events and thus slow growth in throughput.

Table 1. SciPass internal latency

Step	Module	Task	ET (ms)
1	Bro	Detection	20
2	SciPass	Create Rules	4
3	Switch	Install Rules	40
Total			64

6. Reactive Bypass

In the final test we performed another GridFTP transfer between IU and Argonne with SciPass reactively bypassing around the firewall. For these tests, Bro was simply configured to identify data transfer flows in the capture. The implication of this approach is that the bypass will not kick in until flow start time + reaction latency. A likely superior approach for future evaluation involves examining the control channel to identify new data transfer flows and bypassing before or in parallel to the data transfer flows being established, effectively reducing the impact of detection latency on data transfer performance.

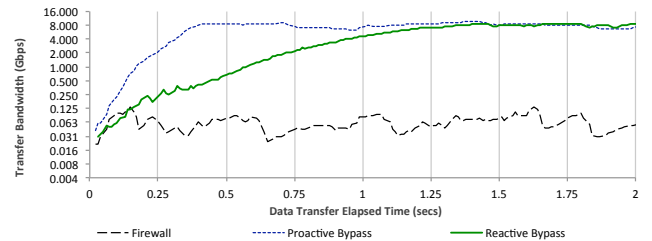


Figure 5 data transfer with SciPass reactive bypass

In this test, the transfer achieved equivalent throughput to the proactive bypass within 1.5 seconds. Over a prolonged transfer there appears to be no sustained transfer performance impact beyond a lower rate of growth in transfer speed. The reactively bypassed flow doubled the average transfer rate of the firewall

path within 250ms. By 250ms into the firewall path based transfer, approximately 1.9Mbytes had been sent.

7. Analysis

The results indicate that least some modern 10Gbps capable firewalls have a significantly negative impact on science data transfers. SciPass is able to detect science data transfers and program alternative forwarding paths in 64ms. Reactively bypassed test transfers achieved double the transfer rate of the firewalled path within 250ms and the same transfer rate as not having a firewall within 1.5 seconds. This indicates that this approach would yield significant performance improvements for any data transfers greater than 2 Mbytes using variable bit rate protocols.

It should also be noted how difficult it was concretely identify the source of packet loss in our lab environment. The lab had nearly ideal conditions, as we controlled every component in the path and the only traffic on the firewall was for our test flow. The cause of this difficulty relates to a lack of exposed counters that indicate when packets are lost in devices like firewalls. In production, the multi-domain nature of the Internet inhibits access to all counters along the forwarding path. Such challenges highlight the need for a key element of the Science DMZ, active measurement and in particular PerfSONAR[9]. Iperf3 was selected as a test tool in part because it is a part of the PerfSONAR system.

8. Future work

SciPass is still under active development, one remaining performance feature is the previously mentioned ability to detect and establish bypass rules for multi-flow protocols like GridFTP. By inspecting the control channel rather than the transfer channel, we hope to further reduce the performance impact of detection latency by starting the detection process before the data transfer.

The effectiveness of SciPass's load balancing routines has not yet been evaluated. As development progresses, we are looking to evaluate against live network traffic on the Indiana University Campus network.

Tests in this paper were performed with both hosts using Hamilton TCP as it is the default for many modern Linux distributions. To better understand the generalized suitability of this approach evaluations of other algorithms is desired.

Beyond transfer performance, this approach has the potential to reduce costs for a number of infrastructure components. A more detailed analysis of the cost impact would aid operators in understanding the suitability of this approach in operations.

9. ACKNOWLEDGMENTS

Our thanks to ESnet for hosting a set of DTN test points and readily accessible performance tuning guides. These resources were very helpful in our evaluations.

Thanks also to Brocade Communication Systems Inc. who provided the switch hardware support and technical input.

10. REFERENCES

- [1] Dart, Eli, Lauren Rotman, Brian Tierney, Mary Hester, and Jason Zurawski. "The science dmz: A network design pattern for data-intensive science." *Scientific Programming* 22, no. 2 (2014): 173-185.
- [2] SciPass <https://github.com/GlobalNOC/SciPass>
- [3] FlowScale
<http://www.openflowhub.org/display/FlowScale/FlowScale+Home>
- [4] McKeown, Nick, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. "OpenFlow: enabling innovation in campus networks." *ACM SIGCOMM Computer Communication Review* 38, no. 2 (2008): 69-74.
- [5] Vern Paxson "Bro: A System for Detecting Network Intruders in Real-Time" *Computer Networks*, 31(23–24), pp. 2435-2463, 1999.
- [6] Allcock, William, John Bresnahan, Rajkumar Kettimuthu, Michael Link, Catalin Dumitrescu, Ioan Raicu, and Ian Foster. "The Globus striped GridFTP framework and server." In *Proceedings of the 2005 ACM/IEEE conference on Supercomputing*, p. 54. IEEE Computer Society, 2005.
- [7] Iperf3 <https://github.com/esnet/iperf>
- [8] R.N. Shorten, D.J. Leith, "H-TCP: TCP for high-speed and long-distance networks" *Proc. PFLDnet*, Argonne, 2004
- [9] Hanemann, Andreas, Jeff W. Boote, Eric L. Boyd, Jérôme Durand, Loukik Kudarimoti, Roman Łapacz, D. Martin Swamy, Szymon Trocha, and Jason Zurawski. "Perfsonar: A service oriented architecture for multi-domain network monitoring." In *Service-Oriented Computing-ICSOC 2005*, pp. 241-254. Springer Berlin Heidelberg, 2005.